

Máster en Ingeniería de Sistemas e Informática

Trabajo Final de Máster

Sistema de Monitorización de Vehículo Eléctrico y Reserva de Punto de Carga

Autor

Juan Almajano Francoy

Director

Enrique Torres Moreno



Universidad
Zaragoza



Escuela de
Ingeniería y Arquitectura
Universidad Zaragoza

RESUMEN

En los próximos años, el sector energético va a experimentar un cambio en la forma de generar, distribuir y consumir la energía eléctrica. Se pretende mejorar el balance entre la producción y demanda mejorando la fiabilidad del suministro y reduciendo costes.

Las denominadas Redes Inteligentes o “ *Smart Grid*” se ocupan de poner en contacto mediante Tecnologías de la Información y las Comunicaciones (TIC) a productores y consumidores en tiempo real.

La inclusión del vehículo eléctrico (VE) agrega la variabilidad en la energía por zonas geográficas debido a las necesidades de movilidad. Ya que la implantación del vehículo eléctrico trae consigo la necesidad de instalar **electrolineras de carga rápida**, esto puede provocar aleatoriedad en la necesidad de energía en la zona donde se encuentran éstas.

El Trabajo Final de Máster (TFM) ha sido realizado gracias a los medios materiales proporcionados por la fundación CIRCE y sin los que yo no podría haber llevado a cabo. Este proyecto ha sido llevado fuera del horario de trabajo que vengo desarrollando en la fundación como informático en proyectos relacionados con la gestión de la recarga de VE.

La motivación que me ha llevado a realizar este TFM fue ver el impacto que el uso del VE realiza en la red eléctrica debido a la necesidad de carga de las baterías que utilizan a modo de depósito de energía. Esto junto con que la fundación CIRCE no estaba realizando ningún proyecto que utilizara el VE como elemento real del sistema.

Se ha creado un prototipo de sistema que monitoriza el estado de carga y permite **planificar** las necesidades energéticas de un vehículo eléctrico disponible en el CIRCE, y comunicarlas con un servidor centralizado que planifica la recarga en electrolineras. El prototipo permitirá disponer de una infraestructura real de bajo coste necesaria para la investigación sobre la inclusión del VE como agente independiente.

Los datos del vehículo han sido extraídos transparentemente a través de un **sistema de monitorización empotrado**. Se ha utilizado un *Smartphone* como interfaz de usuario interconectada con la placa de control inalámbricamente (*WiFi-Bluetooth*). La petición de la reserva se realiza vía **3G** por internet desde el Smartphone, añadiendo información sobre el destino y la posición **GPS** actual. Se han realizado aplicaciones específicas tanto en el sistema de monitorización como en el Smartphone bajo el S.O. **Android**.

El prototipo de muestra la posibilidad de realizar un sistema de monitorización simple y transparente y además económicamente asequible.

ÍNDICE

Introducción	7
Contexto	11
Contenido	12
Arquitectura.....	13
Objetivos y decisiones de diseño	15
Desarrollo Trabajo.....	17
Comunicación VE	18
Sistema Monitorización.....	22
Smartphone.....	25
Conclusiones	27

1 Introducción

En los últimos años se ha venido produciendo un incremento de emisiones de CO_2 a la atmósfera debido en gran parte a liberación de gases contaminantes en el proceso de la generación de energía mediante materiales fósiles. Esto podría llegar a ser un problema si no tomamos medidas para reducir los niveles de gases. Además, estas energías no renovables (carbón, petróleo y gas) se encuentran en la naturaleza de una manera limitada por lo que se hace indispensable conseguir potenciar las energías renovables (viento, sol, geotérmica).

En las centrales de generación de energía eléctrica a partir de materiales como carbón, petróleo o incluso nuclear, podemos regular la cantidad de energía que queremos producir. No es así en la mayor parte de instalaciones de energías renovables que pese a ser una fuente inagotable de energía necesitan de unas condiciones climatológicas específicas para poder producir energía de forma predecible (aerogeneradores->viento, fotoeléctrica->sol).

Se intenta que la curva de producción de energía sea lo más parecido a la de la demanda. Cuando la producción de energía supera a la demanda se produce el problema de qué hacer con la energía sobrante ya que la energía tiene que ser consumida al momento o se pierde y no puede ser vendida. También se produce un problema cuando la demanda supera a la producción, dejando huecos en la red eléctrica. Si la curva de producción/demanda alcanza un mínimo se denomina momento “*valle*” y si al contrario alcanza un máximo se denomina “*pico*”.

En la siguiente gráfica podemos observar la diferencia entre la demanda de energía y la producción de energía eólica. El eje X de coordenadas representa un intervalo de tiempo de una semana y en el eje Y de coordenadas podemos observar la energía generada en MegaWattios (MW).

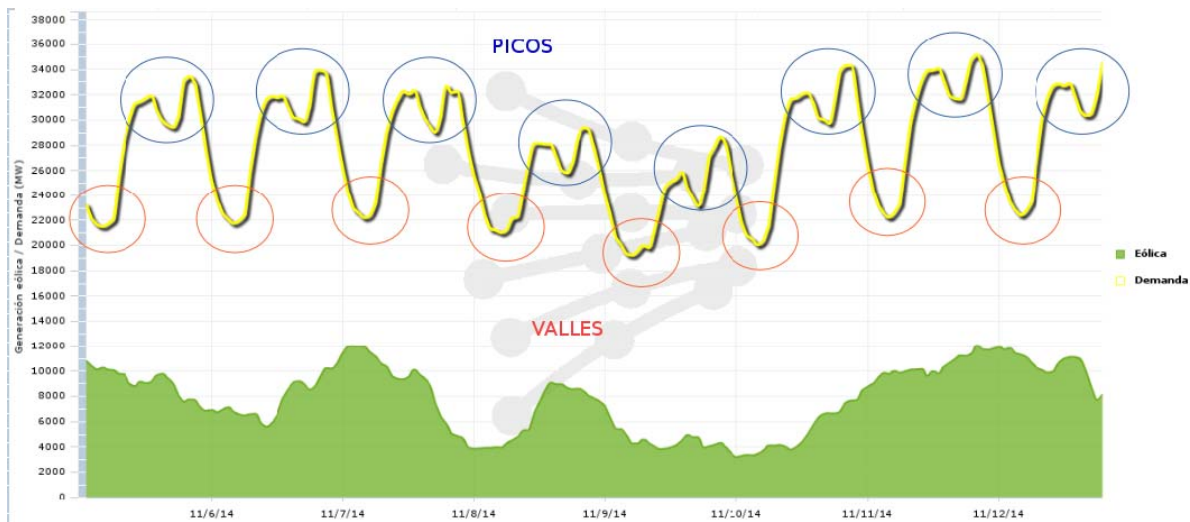
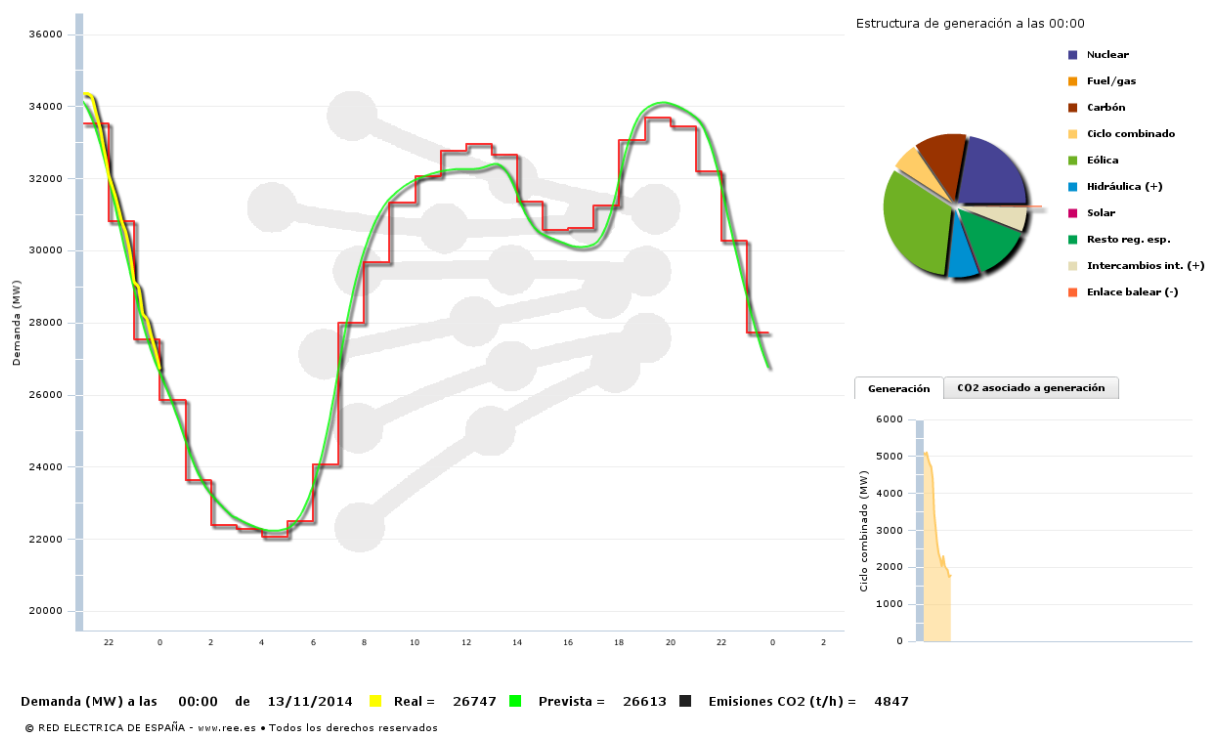


Ilustración 1: fuente: Red Eléctrica Española www.ree.es

La curva representada en amarillo muestra la demanda de energía en la línea del tiempo. Se puede observar un patrón repetitivo entre días. Se observan ocho zonas valle y ocho zonas pico con valores variantes inter e intra-día. La curva representada en color verde muestra la energía generada mediante producción eólica a lo largo de la semana. La gráfica muestra de forma clara la aleatoriedad en los distintos momentos de la producción y la no coincidencia con la demanda. La diferencia, que se observa en la gráfica, de energía entre la demanda y la producción debe ser cubierta con otro tipo de fuentes de energía ya sean renovables como solar, térmica hídrica u otras de carácter no renovable como las de ciclo combinado o nuclear.

En los próximos años vamos a experimentar un gran cambio en la forma de producir y consumir la energía, estamos obligados a conseguir una gestión de energía lo más eficiente posible y eso significa poder actuar sobre la red en tiempo real. Hoy en día se realiza la previsión energética a un día vista.

A continuación observamos una gráfica en la que podemos observar la previsión de la demanda que se realiza a un día vista. El eje X de coordenadas nos muestra el intervalo temporal de la previsión por horas. El eje Y muestra la energía en MW. En la parte derecha observamos la generación por volumen de producción y categorías.



La curva representada en color verde muestra la predicción de energía, mientras que la curva amarilla muestra la generación de energía en tiempo real. Se puede apreciar la gran diferencia de consumo de energía a diferentes horas del día (picos y valles). Se intenta ajustar lo máximo la curva entre generación y producción mediante la predicción de la necesidad de energía. En el proceso intervienen distintas fuentes de energía y multitud de suministradores distintos. Si la energía producida es menor que la demandada se crean huecos y se pierde fiabilidad que conlleva quejas de los usuarios y pérdidas de oportunidades de negocio. Si ocurre lo contrario la energía se pierde, dada la dificultad de almacenarla, y por lo tanto se pierde eficiencia aumentan los costes.

A la vez que se introducen nuevas formas de generar y almacenar la energía de una manera más distribuida, la complejidad del sistema aumenta. Para lo que se necesita mejorar el control en el proceso para que haya un suministro seguro y de calidad. Aparece un concepto que se denomina “Red Inteligente” o en inglés “Smart Grid” que se podría definir como el uso de tecnologías de la información y comunicación (TIC) entre productores y consumidores. Mediante elementos de monitorización y control en tiempo real se puede además conseguir una mayor integración medioambiental y poder dar accesibilidad a los distintos agentes al sistema.

En la siguiente ilustración se muestra un ejemplo de una Smart Grid. En la imagen podemos observar la interconexión TIC entre los distintos agentes que participan en la producción y el consumo de energía.

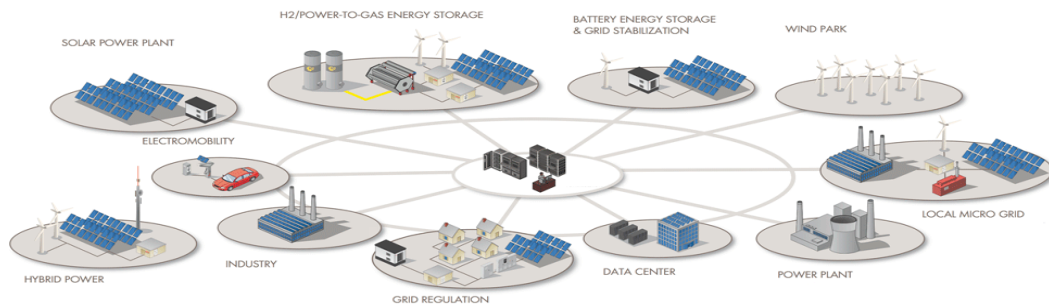


Ilustración 2: fuente: AEG POWER SOLUTIONS www.aegps.com

La gestión se podría realizar de manera centralizada como se muestra en la figura, aunque es común también crear pequeñas micro redes que pueden actuar a la vez como productores y consumidores. Etiquetado como *electromobility* se observa un punto de recarga de Vehículos Eléctricos (VE).

Los motores eléctricos de los Vehículos se pueden considerar que están bastante desarrollados, no es así con las baterías que son aún ineficientes, pesadas, limitadas tanto en capacidad de energía como en el número de ciclos de carga-descarga. Estas limitaciones incrementan la necesidad de crear numerosos puntos de carga bien distribuidos y rápidos. Además por otro lado la instalación de puntos de recarga rápida impactará en las necesidades de energía de la red, creará picos móviles en la red y por tanto se incrementará la incertidumbre en la necesidad de energía sobre la red eléctrica. Se incentiva así una aleatoriedad en el espacio y en el tiempo. Si tenemos en cuenta que en nuestros días un punto de recarga de un VE puede necesitar una potencia de unos 10 kW (Tesla está empezando a fabricar cargadores de 120 kW), por lo que a medida que aumente el parque de VE se incrementa el impacto sobre la red eléctrica.

Una característica del VE es la posibilidad del novedoso uso del mismo como un acumulador-suministrador de energía móvil. Pudiendo revender su energía almacenada a otros coches o a la red en el rol de generadores de energía. Sabiendo tus necesidades y costumbres de uso se puede realizar una previsión, disponiendo de numerosos puntos de recarga móviles en cualquier

parking, aportando energía a otros VE o a un edificio. Esto forma parte de mi futura investigación en la fundación CIRCE, no siendo objetivo de este TFM.

A continuación se muestra un ejemplo de la parte de la red inteligente en la que nos queremos centrar. Por un lado observamos las electrolineras conectadas a la red de distribución eléctrica. Por otro lado tenemos VE sobre los que se monitoriza el estado de las baterías y comunican con el servicio centralizado sus necesidades. El servicio planificará las recargas en las electrolineras. Todo ello interconectado por internet a través de distintas tecnologías.

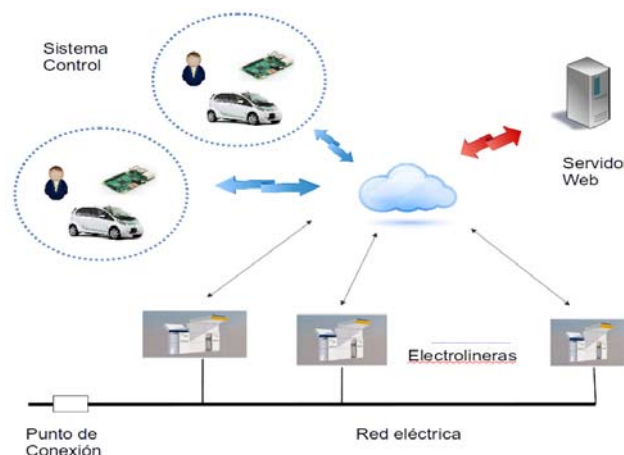


Ilustración 3

El funcionamiento básico consiste en: el sistema de control del VE solicita una reserva de carga al servidor enviando los datos de estado de la batería del coche, la posición GPS y el destino. El servidor web comprueba las electrolineras disponibles en el camino al destino. Una vez situado el VE en la electrolinera, ésta se encarga de tomar o volcar energía al Punto de Conexión con la red.

El concepto de redes inteligentes introduce la idea de poder realizar una comunicación continua y bidireccional entre los diferentes agentes involucrados en la generación y consumo de energía, para lo que se tiene que crear una plataforma específica.

1.1 Contexto

Desde Enero de 2013 estoy trabajando como Ingeniero Técnico Informático en la fundación Centro de Investigación de Recursos y Consumos Energéticos (CIRCE) que tiene como objetivos el impulso de la mejora de la eficiencia energética y el despliegue de energías

renovables mediante el desarrollo de actividades de I+D+i y acciones formativas.

Comencé con una beca de iniciación a la investigación y en Octubre he pasado a ser contratado. CIRCE está involucrado en proyectos relacionados con la carga de VE (Zem2All, SIRVE) donde vengo realizando diversos desarrollos informáticos relacionados con la gestión de los puntos de recarga rápida.

Fue el hecho de que no se estuviera realizando ningún proyecto agregando el VE como agente participativo el que me hiciera decidir a investigar sobre el tema, desarrollándola aparte de la actividad que desarrollo en CIRCE. Este trabajo se ha realizado fuera del horario laboral, adquiriendo responsabilidad plena en todos los pasos del proceso.

Dentro de este marco y gracias a las facilidades proporcionadas por CIRCE tanto en infraestructura como en materiales, he realizado mi proyecto.

Desde el mes de Julio vengo realizando las diversas tareas relacionadas con este proyecto.

1.2 Contenido

Este documento consta de cinco capítulos en el que se ha aportado con una solución en el escenario propuesto, así como la realización de pruebas y la validación de todos los elementos.

El primer capítulo de la memoria ofrece un Introducción y sitúa el TFM en su contexto.

En el segundo capítulo se establece los objetivos para la realización de este trabajo, a la vez que se evalúa las decisiones de diseño tomadas.

El tercer capítulo desarrolla el proyecto, explicando los pasos realizados, las soluciones adoptadas y las pruebas realizadas. Realizando las pruebas necesarias para la validación del sistema.

El cuarto detalla las conclusiones a las que se ha llegado tras la realización del TFM.

1.3 Arquitectura

El proyecto cuenta con un Vehículo Eléctrico modelo **Mitsubishi i-Miev** disponible en las instalaciones de la fundación CIRCE. Queremos monitorizar sus parámetros por medio de un sistema conectado al vehículo de forma transparente. Como interfaz de usuario se va a utilizar un Smartphone aprovechando la pantalla táctil, la conexión inalámbrica (WiFi y Bluetooth), la posibilidad de conexión a internet y el receptor GPS del que dispone. El sistema se conecta con un servidor externo el cual se comunica con los distintos agentes del sistema.

Las principales características del Mitsubishi i-Miev utilizado para las pruebas son:

Potencia Máxima	67 Cv (49 Kw) / 2.500 r.p.m. - 8.000 r.p.m.
Par motor máxima	180 Nm / 0 r.p.m. - 2.000 r.p.m.
Potencia nominal Cv (Kw)	48 Cv (35 Kw)
Tipo de batería	Baterías de ión Litio
Voltaje total	330 V
Energía total	16 Kwh
Carga convencional	16 A: 6 horas aproximadamente
Carga rápida	Protocolo CHAdeMO: 30 minutos aproximadamente
Velocidad máxima (km/h)	130 km / h
Consumo de energía eléctrica	135 Wh / km
Autonomía (km)	150 km

Ilustración 4

Destacamos la capacidad de la batería que es de 16 Kwh, la cual tardaría seis horas aproximadamente en tener plena carga si la cargáramos en un enchufe normal como los que disponemos en casa, mientras que si la realizamos con carga rápida tarda alrededor de 30 minutos. Otra característica es el consumo de energía de 135 Wh/km que permite una autonomía de 150 km.

La siguiente figura presenta un esquema de la arquitectura implementada:

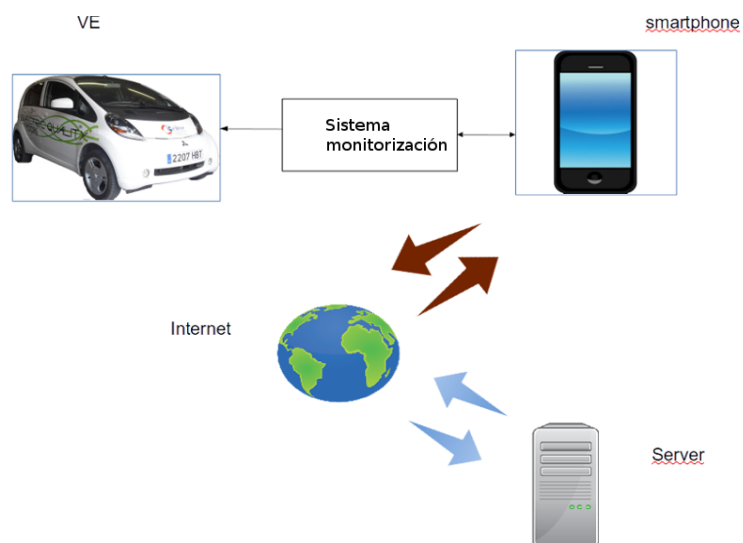


Ilustración 5

En la figura se muestra la comunicación que se realiza entre los distintos dispositivos. El sistema de monitorización recoge los datos del VE. El Smartphone recoge estos datos y junto a la posición actual por GPS y el destino o uso programado se enviará al servidor.

2 Objetivos y decisiones de diseño

El objetivo de este TFM es implementar un prototipo que monitorice el estado de la batería de un Vehículo Eléctrico Mitsubishi i-Miev, estableciendo una infraestructura de comunicación con un servidor.

Para ello se han establecido los siguientes objetivos:

- Monitorizar de forma transparente los datos del vehículo. Para lo que se tiene que:
 - Diseñar un prototipo de comunicación
 - Validar la comunicación usando un analizador de protocolo.
 - Implementar el software del sistema empotrado de monitorización
- Implementar interface de usuario y aplicación de control.
- Montar servidor con funcionalidad mínima para verificar el sistema.

Hemos diseñado un prototipo de la placa que va a realizar la comunicación, montándolo en un *protoboard* para realizar las pruebas. Para la validación física se ha utilizado un osciloscopio y para la validación lógica se ha utilizado un analizador de protocolo y una herramienta de análisis de tramas. Una vez validada la comunicación se ha implementado en un circuito integrado, volviendo a validar la comunicación.

Como placa de control se ha buscado un diseño que pudiera estar lo más integrado posible con la centralita del vehículo y que la implementación fuera lo más sencilla posible. En principio se barajó entre las opciones de trabajar con NanosG20, Raspberry PI o Arduino. Finalmente, la decisión fue la de utilizar Raspberry PI B+. Éste dispone de sistema operativo lo que facilita la instalación de librerías y el uso de herramientas, resultando a su vez económica.

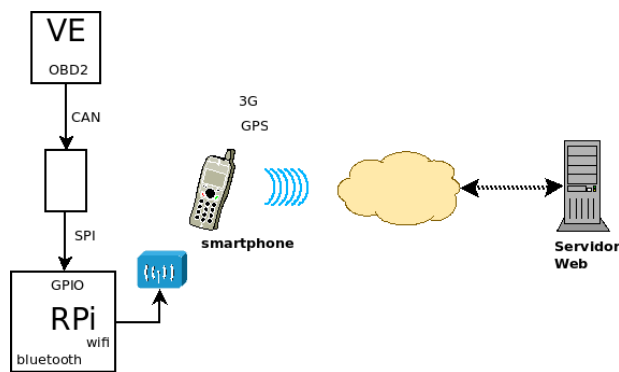
En cuanto a la interfaz de usuario se ha decidido utilizar un Smartphone porque dispone de pantalla táctil, procesador, conexión a internet y GPS. Otro motivo es la generalización de estos dispositivos y su fácil disponibilidad por parte de cualquier usuario, por lo que implica un bajo coste añadido. Para realizar la comunicación entre el sistema de monitorización y el Smartphone se ha realizado una comunicación inalámbrica porque es la más conveniente y cómoda de cara al usuario. He planteado dos opciones. Una ha sido la conexión por WiFi entre ambos dispositivos, por otro lado se ha realizado una comunicación por Bluetooth junto a una aplicación sobre el S.O. Android utilizando el plugin SDK de Android para Eclipse

La lógica de negocio en el servidor de aplicaciones se ha implementado de una manera meramente presencial ya que se considera que la comunicación entre servidor y punto de carga para la reserva no es objetivo de este trabajo.

3 Desarrollo Trabajo

En el siguiente capítulo se va a explicar en detalle el trabajo que se ha llevado a cabo.

A continuación se muestra un esquema de la comunicación del trabajo realizado.



La Raspberry Pi monitoriza el estado del VE a través del módulo que transforma la comunicación SPI a CAN. El smartphone toma los datos de la Raspberry Pi a través de una comunicación wifi y los envía a través de internet a un servidor.

A continuación se explicará como se ha ido desarrollando el proyecto y como se ha implementado, verificado cada elemento del sistema.

3.1 Comunicación VE

Para poder implementar el sistema es imprescindible poder monitorizar los parámetros del VE de forma transparente. Se barajaron distintos métodos para la adquisición de datos como la conexión directa a la batería, monitorizando mediante la conexión al bus de comunicación de la centralita del coche o de la maniguera. Se ha elegido la segunda por ser la manera más eficaz de extraer los datos. El sistema OBD (On Board Diagnostics) es un sistema de diagnóstico del vehículo, aportando información muy detallada sobre elementos del vehículo.

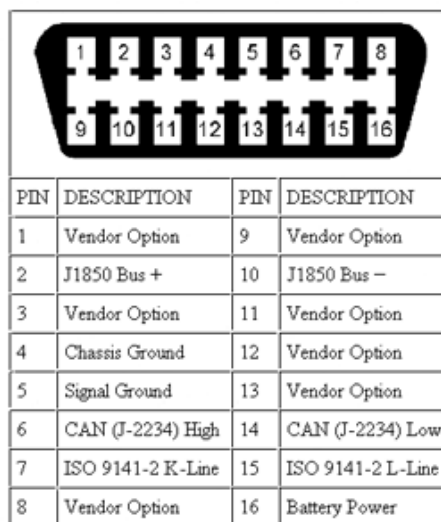
El vehículo dispone de una centralita electrónica ECU (Electronic Control Unit) y diversos dispositivos que se comunican entre sí.

El protocolo CAN (Controller Area Network) es un protocolo de comunicación para redes distribuidas con necesidades de tiempo real. Se caracteriza por una comunicación mediante prioridad de mensajes. Cualquier dispositivo electrónico de un vehículo (frenos, inyección, airbag, etc) puede comunicarse entre sí. La comunicación fiable y robusta se realiza mediante un bus multiplexado que permite altas velocidades incluso en entornos difíciles.

Las velocidades pueden llegar hasta 1 Mbps. Su utilización es sobre todo en vehículos pero también se utiliza en industria.

La norma SAE J2284 estandariza la tipología de conector utilizado en la actualidad, define la versión específica de CAN bus utilizada en el conector OBD-II.

El siguiente dibujo muestra el esquema de un conector OBDII



OBD-II Connector and Pinout

Ilustración 6

En la figura se observan los pines necesarios para la comunicación con el VE son el 14 (Can Low) y el 6 (Can High).

La siguiente figura muestra una arquitectura de CAN bus en un vehículo eléctrico.

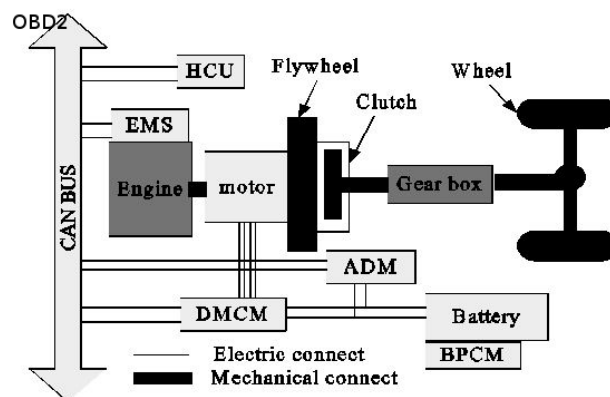


Ilustración 7. fuente: intech url:<http://www.intechopen.com>

Este es un esquema en el que todos los dispositivos están conectados a un mismo bus. El módulo que nos interesa monitorizar es el de la batería. Los datos se extraen a través del puerto OBD2.

Con la colaboración de Javier Bergeles, que trabaja como investigador en el área IER (Investigación Energías Renovables) de la fundación CIRCE, se ha diseñado un circuito que permite realizar la comunicación.

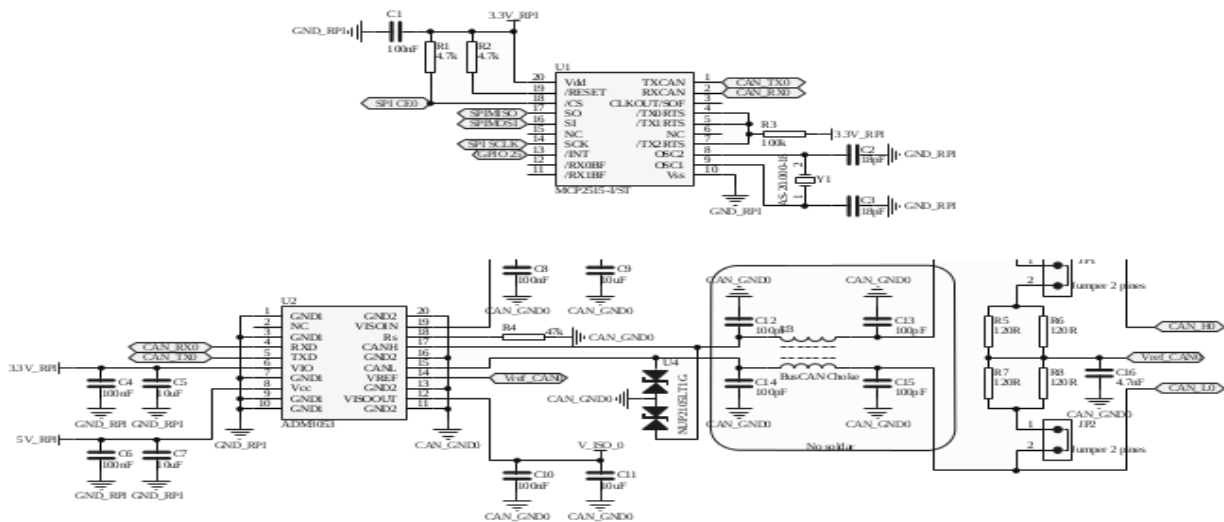


Ilustración 8

Este dibujo representa el esquema del circuito utilizado. El módulo que realiza la conversión SPI-CAN es el módulo **mcp2515** de *Microchip*, que se conecta a los pines SPI de la GPIO de la Raspberry pi. El módulo **ADM 8053** se encarga de transformar las señales CAN Tx y CAN Rx en CAN Low y CAN High.

Una vez diseñado el esquema se ha implementado en un *protoboard* para poder evaluar su funcionamiento.

La figura muestra el esquema implementado en un *protoboard*.

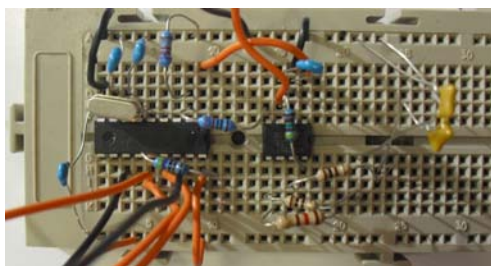


Ilustración 9

Para la validación del protocolo se ha dispuesto de un analizador CAN HD67216 y del software CAN Analyzer disponible en <http://www.adfweb.com>. Para verificar el protocolo se han realizado varias pruebas conectando además un osciloscopio. Se han enviado varios mensajes analizando y verificando el correcto funcionamiento del protocolo.

En la imagen de la izquierda se muestra el analizador de CAN, a la derecha se muestra la

salida observada en el osciloscopio, el mensaje “178#deadbeef”.



Ilustración 10

A continuación se muestra una trama típica de CAN extendida:

SOF	ID	CONTROL	DATOS	CRC	ACK	EOF
1 bit	32 bits	6 bits	64 bits	16 bits	2 bits	7 bits

SOF: bit de inicio de trama.

ID: 29 bits de identificación, 1 bit para definir el mensaje como trama extendida un bit SRR y un bit RTR.

CONTROL: 1 bit que indica trama extendida (RB1), 1 bit reservado (RB0) y 4 bits para definir la longitud de datos (DLC).

DATOS: Nº de bytes definidos en DLC.

CRC: 15 bits y un delimitador CRC. Para detección de errores.

ACK: Indica la recepción correcta del mensaje.

EOF: bits de fin de trama.

Una vez testeado el *protoboard* se ha implementado el esquema en una placa de circuito integrado ya que es poco fiable trabajar con el montaje sobre el *protoboard*, debido a la cantidad de componentes y conexiones que dispone. Utilizando el programa Altium se ha diseñado el circuito dando como resultado los ficheros del diagrama del circuito que se enviarán posteriormente a la impresora de circuitos.

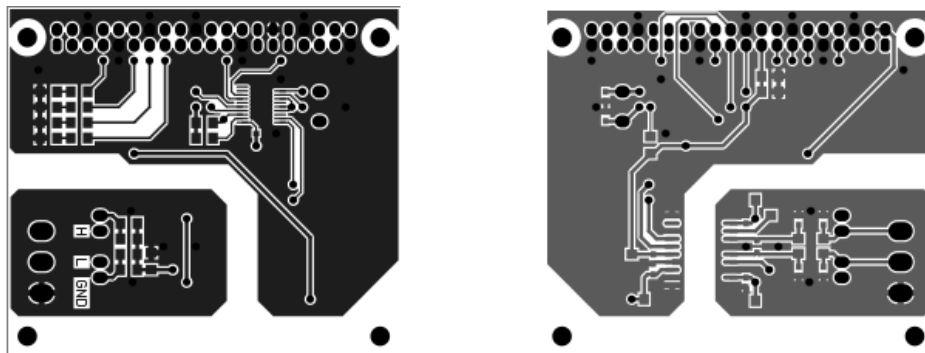


Ilustración 11

En la imagen se muestra el mapa del circuito por la parte de arriba(izda) y de abajo(dcha).

Una vez obtenido los archivos se pasan a una impresora para la realización física del circuito. Como curiosidad, en la imagen se muestra la impresora utilizada para realizar la placa integrada.

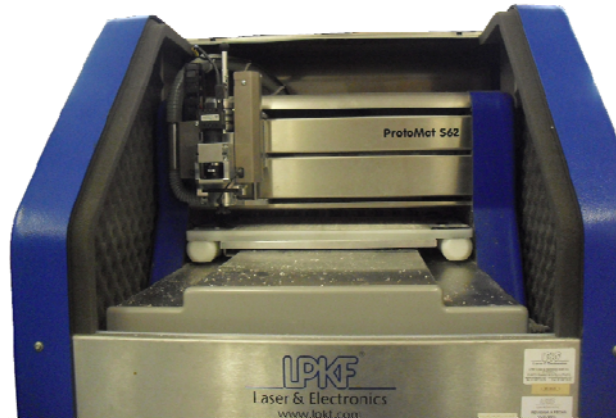


Ilustración 12

Se han realizado las mismas pruebas en el circuito integrado que las que se realizaron con el protoboard para analizar y verificar el correcto funcionamiento

Como resultado hemos obtenido una placa integrada que realiza una comunicación entre SPI y CAN.

3.2 Sistema Monitorización

Para realizar la monitorización del sistema se ha dispuesto de una Raspberry PI modelo B+. Implementado en Julio de 2014, es una revisión del modelo B. Respecto al modelo anterior aumenta el número de puertos USB a cuatro y el número de pins de GPIO (General Purpose Input/Output) a 40. Se ha elegido este controlador por la disponibilidad de este en el mercado, habiendo abundante información sobre el componente en internet y por el precio.

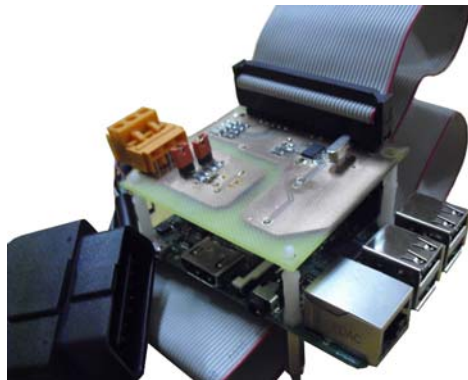


Ilustración 13

En la figura se muestra la Raspberry Pi conectada mediante un bus al circuito integrado

fabricado anteriormente.

Se ha instalado el sistema operativo Raspbian (Version Septiembre 2014, 3.12.28) basado en Debian. Este es un sistema libre optimizado para usar en Raspberry Pi. Destaca por la gran cantidad de paquetes (más de 35.000) dispuestos para una fácil instalación. Su desarrollo está soportado por la comunidad.

La comunicación entre el VE y la Raspberry PI se realiza a través del componente fabricado en el apartado anterior. Se necesitan instalar y configurar unas librerías y módulos extras para que funcione correctamente. Tanto el software como la guía de instalación se pueden descargar de la url: "<http://lnxpps.de/rpie/>".

Una vez instalado y configurado el software se procede a comunicar con la ECU y monitorizar la información del VE.

A continuación se muestra como se ha conectado la Raspberry Pi con el VE a través del puerto OBD2 durante las pruebas.

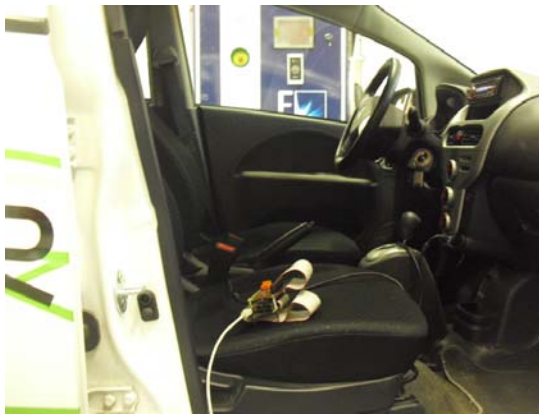


Ilustración 14

La comunicación durante las pruebas se ha realizado por puerto ethernet, se ha tomado esta decisión para poder conectar remotamente a la RPi. Para alimentar la Rpi se ha conectado a través de un cargador de coche y el puerto CAN se ha conectado al puerto OBD del VE que se encuentra debajo del volante.

Conectando la Raspberry Pi al VE a través de hardware implementado se han realizado varias pruebas de comunicación.

A continuación un ejemplo de las pruebas realizadas

```
pi@raspberrypi ~ $ candump can0
can0 05FBBB73 [8] B0 B0 79 CF 0D 46 00 16
can0 033936DA [8] 05 00 00 00 40 00 00 00
can0 05FBB949 [8] F1 7F 00 FF 19 80 01 BE
can0 05FBBB46 [8] 29 14 2E 20 20 00 00 19
can0 03393308 [8] 00 03 E8 00 00 00 00 00
can0 05FBB374 [8] 36 31 B7 FE 4C 48 49
can0 05FBBEE3 [8] 04 47 47 00 01 60 01 60
can0 05FBBEE4 [8] 04 00 00 00 01 60 01 60
can0 03393285 [8] 07 D0 14 00 90 FE 0C 10
can0 05FBBA00 [8] 00 03 C0 00 C0 00 FF FF
can0 05FBBA08 [8] 00 20 60 00 C0 00 C0 00
can0 03393288 [8] 07 D0 27 10 AA 00 11 10
can0 05FBBB73 [8] B0 B0 79 CE 0D 46 00 16
can0 05FBBB25 [2] 01 00
can0 03393418 [7] 50 00 00 00 00 00 00
can0 05FBBA10 [7] 00 00 00 00 80 00 00
```

...

Las tramas son del tipo CAN extendido. Se muestra los campos puerto “can0”, el identificador que corresponde a 29 bits, por ejemplo, “05FBBA10”, el tamaño del array “7” y el array de datos de 8 bits “00 00 00 00 80 00 00” en hexadecimal.

El identificador de la trama que contiene la información relativa al estado de carga de la batería es 374. Para calcular el valor exacto hay que realizar unas operaciones entre los datos contenidos en el array de datos recibido en la trama. En este caso el valor es:

$$\text{Batt_soc} = (\text{array_datos}[2] - 10) / 2$$

Para solucionar la conexión inalámbrica por wifi, se ha añadido un periférico para dar acceso, en concreto se ha dispuesto de un CS L-lápiz modem USB WLAN 300 Mbits/s. Ha sido necesario instalar el driver “rt1871xdrv” para la comunicación wifi, el código necesario está alojado en la url: http://www.adafruit.com/downloads/adafruit_hostapd.zip.

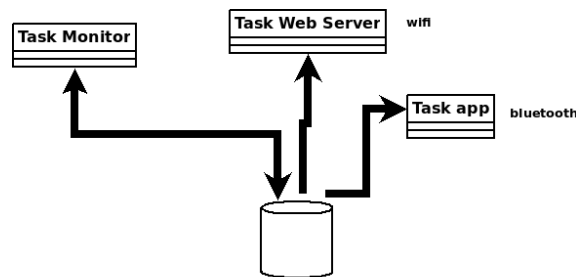
Para solucionar la conexión inalámbrica por bluetooth se ha añadido un adaptador ASUS USB-BT400 para la comunicación bluetooth. Para la instalación de las librerías necesarias se ha seguido un manual disponible en la url: “<http://www.diverteka.com/?p=1880>”.

Para la implementación de la aplicación contenida en la Rpi que se ha desarrollado como parte del TFM se han instalado las herramientas de compilación GNAT y la librería libaws 2.0 necesaria para crear el servidor web. Se ha diseñado e implementado una aplicación

Ada. Se ha elegido este lenguaje por que soporta necesidades de tiempo real, ya que queremos que este monitorizando el estado del vehículo continuamente y a la vez tiene que ser capaz de poder servir esos datos al smartphone.

Una característica de Ada es el uso de tareas (task) que es la manera de ejecutar código secuencial de manera paralela.

A continuación se muestra el esquema de la aplicación,



La aplicación principal ejecuta varias tareas, una que monitoriza el estado de la batería y lo almacena. Otra, un servidor web que pone a disposición los parámetros almacenados a través de la conexión wifi mediante de una página web usando la librería Extjs 4.2 disponible en [“http://www.sencha.com/products/extjs/download/ext-js-4.2.0/2142”](http://www.sencha.com/products/extjs/download/ext-js-4.2.0/2142) que está basada en javascript. Y una última que sirve los datos mediante bluetooth a una app android en el smartphone.

3.3 Smartphone

Se ha elegido un smartphone como interfaz de usuario ya que en sí nos resuelve muchas de nuestras necesidades como pantalla táctil, la conexión a internet y GPS. Y suponiendo que el usuario ya lo tiene, tiene un bajo coste añadido al prototipo.

Para realizar el prototipo se ha dispuesto de un Smartphone Alcatel one touch 6012X con versión firmware 4.2.2 y versión kernel 3.4.5.

Primero se ha creado una red wifi seleccionando como punto de acceso (AP) la Raspberry Pi pero se ha desestimado porque no he conseguido tener acceso de manera dual a internet. Finalmente se ha configurado el smartphone como AP y además como puerta de enlace para conectarse a Internet.

Si utilizamos la red wifi como forma de conectar el sistema con el smartphone, se ha decidido utilizar el navegador web (Mozilla/5.0 AppleWebKit/534.30 Version/4.2 Mobile Safari/534.30) como interfaz de usuario y como controlador realizando peticiones a un servidor. La aplicación ejecuta código javascript y html.

Para la comunicación por blue tooth se ha diseñado una aplicación Android, utilizando la herramienta eclipse con el módulo de desarrollo Android sdk tool rev 23.0.5 de google. La aplicación se ha pensado con el mismo escenario que con la comunicación por wifi.

Se ha realizado el prototipo con una visualización simple ya que no es objetivo de este trabajo el realizar una aplicación final. La aplicación se ha pensado para un determinado escenario en el que un usuario tiene que ir al trabajo todos los días y en las inmediaciones de este se encuentra una electrolinera. Cuando sale de casa el usuario indica a la aplicación el trayecto a realizar, el sistema envía el estado de la batería y la posición GPS realizando una petición a un servidor que es el que realiza la reserva del punto de carga. El usuario cuando llega a la electrolinera está disponible para su uso.

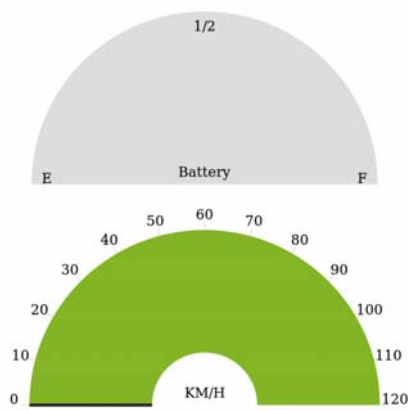


Ilustración 15

En la figura se puede observar los indicadores de batería y velocidad que se muestran en la aplicación realizada.

4 Conclusiones

En el futuro se quiere crear una plataforma en la que se combine varios de los elementos que actúan en la generación, distribución, almacenamiento, consumo de la red eléctrica y las tecnologías de la información. Pudiendo sacar datos de cómo se comporta de manera realista. La inclusión del VE como elemento de una red inteligente va a ser algo fundamental si queremos conseguir una mejora en la fiabilidad del suministro.

Este trabajo es el punto inicial para posteriores investigaciones proporcionando una plataforma en la que poder desarrollar futuros proyectos relacionados con el VE. El uso del VE como agente generador y suministrador (**prosumer**) en el sistema. Cualquier VE, como almacén de energía, podrá revenderla a otros usuarios si no la necesita.

El proyecto realizado demuestra la posibilidad de integración del vehículo eléctrico dentro de una red inteligente de una manera simple y económica.

Se han adquirido nuevos conocimientos en electrónica. Partiendo de un esquema y de los *datasheets* del chip de protocolo. Se ha implementado y fabricado una placa de circuito integrado haciendo uso de software de diseño de circuitos. El circuito fue validado y verificado mediante el montaje previo sobre un *protoboard*, y posteriormente el circuito final fue validado haciendo uso de herramientas de laboratorio como puede ser un osciloscopio y otros elementos de medición.

Se ha analizado el protocolo de comunicaciones con la centralita del VE utilizando un analizador de protocolo y el software correspondiente, realizando aplicaciones de prueba para validar el cumplimiento del protocolo y extraer la información relativa a la batería del VE de forma totalmente transparente.

Se ha instalado el software y creado las aplicaciones sobre el sistema empujado de monitorización basado en la Raspberry PI, para el control, adquisición de la información y comunicación con el Smartphone. La información se sirve vía servicio web si la comunicación es por WiFi, permitiendo acceder a ella desde cualquier navegador http, y mediante aplicación específica si la comunicación es vía Bluetooth.

Se ha creado la interface de usuario del sistema usando un Smartphone para hacer uso tanto de la pantalla táctil como del GPS como de la comunicación móvil por 3G. La inclusión del Smartphone del usuario permite facilitar el uso y la curva de aprendizaje. El coste añadido en el sistema es mínimo si suponemos que el usuario ya dispone de dicho dispositivo.

Los objetivos del TFM se han cumplido satisfactoriamente y en los plazos requeridos. Con el desarrollo de este trabajo se ha comprobado la posibilidad de realizar un sistema de comunicación en tiempo real integrado, transparente, fácil de usar y económicamente viable.